Cosmic Microwave Background Data Analysis

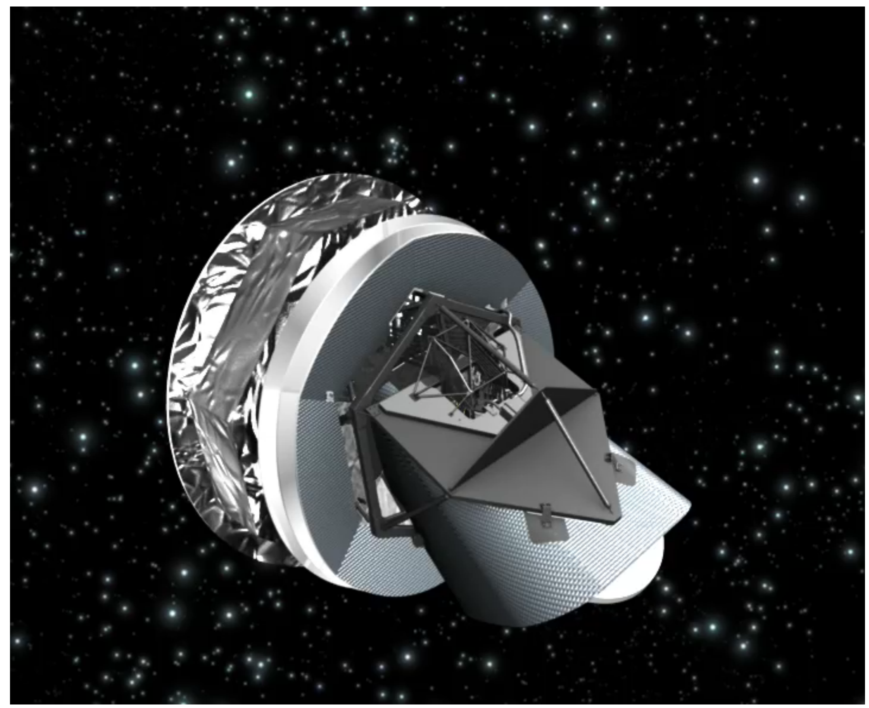
Julian Borrill

Computational Cosmology Center, LBL

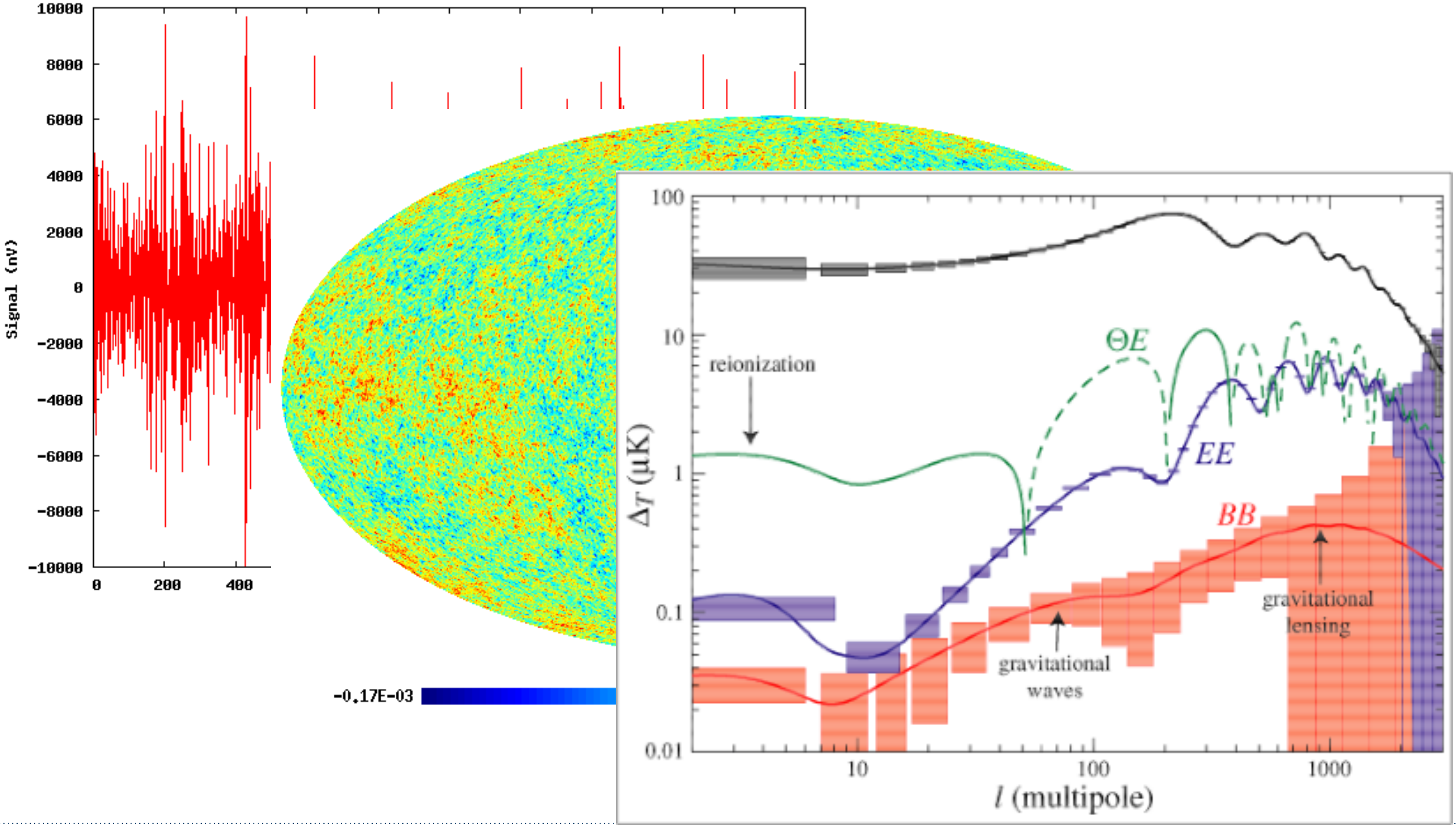
& Space Sciences Laboratory, UCB

CMB Data

- Looking for the wallpaper not the furniture.
- Scanning rather than pointed observations.
- Data components are separately correlated
  - Noise in time domain
  - Foregrounds in pixel domain
  - CMB in multipole domain
- Entire data set is a single data object
  - No divide & conquer approach
  - MTBF issues!
- No database of objects/images, just a handful of maps.

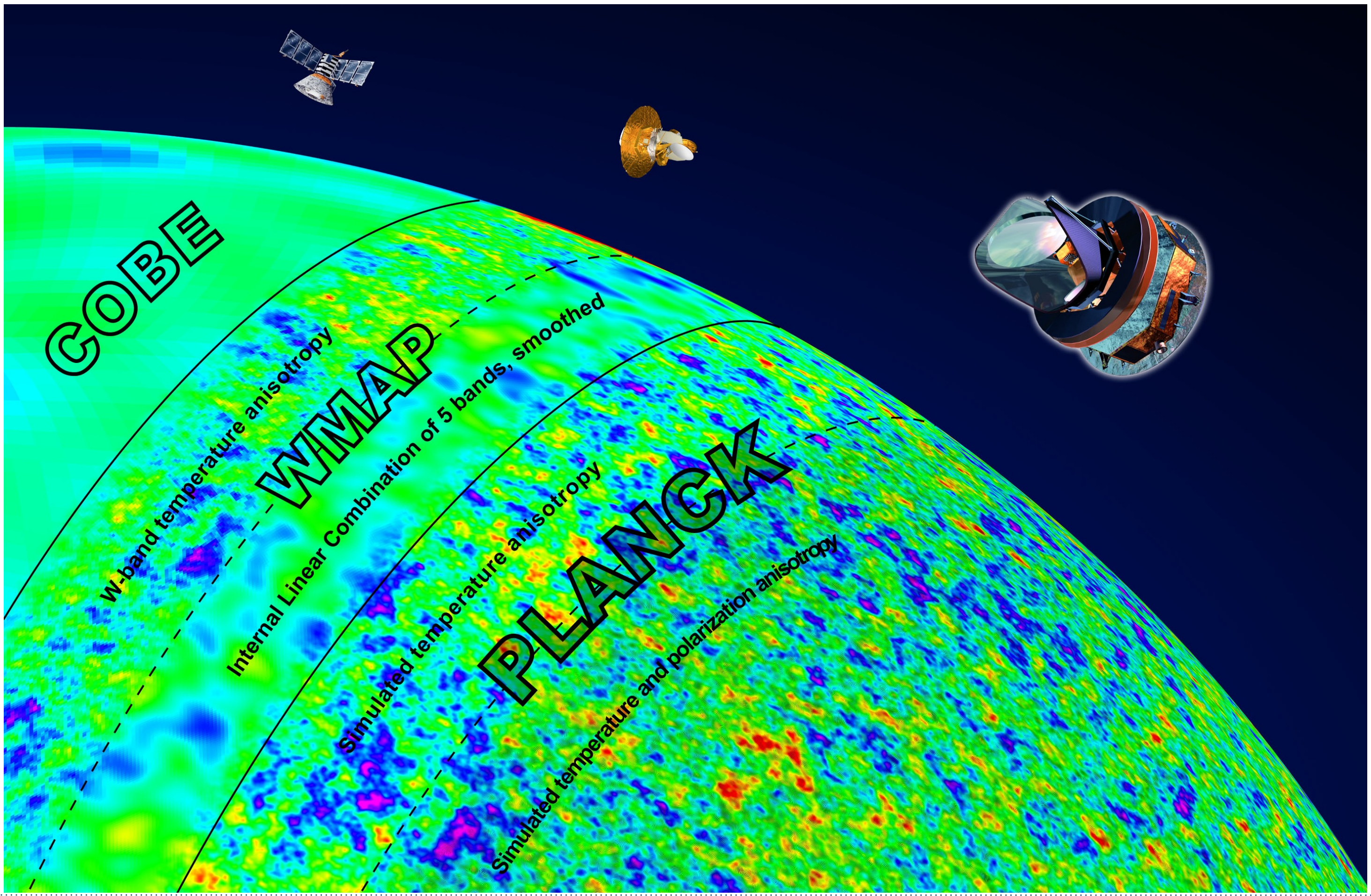


Analysing The CMB



CMB Satellite Evolution

Evolving science goals require (i) higher resolution & (ii) polarization sensitivity.



The CMB Data Challenge

- Extracting fainter signals (polarization, high resolution) from the data requires:
  - larger data volumes to provide higher signal-to-noise.
  - more exacting analyses to control fainter systematic effects.

Experiment	Start Date	Goals	N_t	N_p
COBE	1989	All-sky, low res, T	10^9	10^4
BOOMERanG	1997	Cut-sky, high-res, T	10^9	10^6
WMAP	2001	All-sky, mid-res, T+E	10^{10}	10^7
Planck	2009	All-sky, high-res, T+E(+B)	10^{12}	10^9
PolarBear	2012	Cut-sky, high-res, T+E+B	10^{13}	10^7
QUIET-II	2015	Cut-sky, high-res, T+E+B	10^{14}	10^7
CMBpol	2020+	All-sky, high-res, T+E+B	10^{15}	10^{10}

- 1000x increase in data volume each over past & future 15 years
  - need linear analysis algorithms to scale through 10 + 10 M-foldings !

Computational Challenge

- Data volume drives us to (log-)linear algorithms
  - FFT, SHT, PRNG, sparse MV & Monte Carlo everything.
  - Minimal data reuse (Level 1) so no room to hide non-calculation costs
- Hierarchy of costs (time, power)
  - Data transfer/staging > I/O > Communication > Calculation
- Cost per byte/flop decreases with time/concurrency but *ratios* get worse.
- HPC systems are increasingly heterogeneous & hierarchical
  - keeping up with Moore gets harder and harder.
  - compilers/libraries aren't the (whole) answer.

Keep data as close to the cycles as possible.

Replace IO with communication, communication with calculation.

CMB Data Analysis Evolution

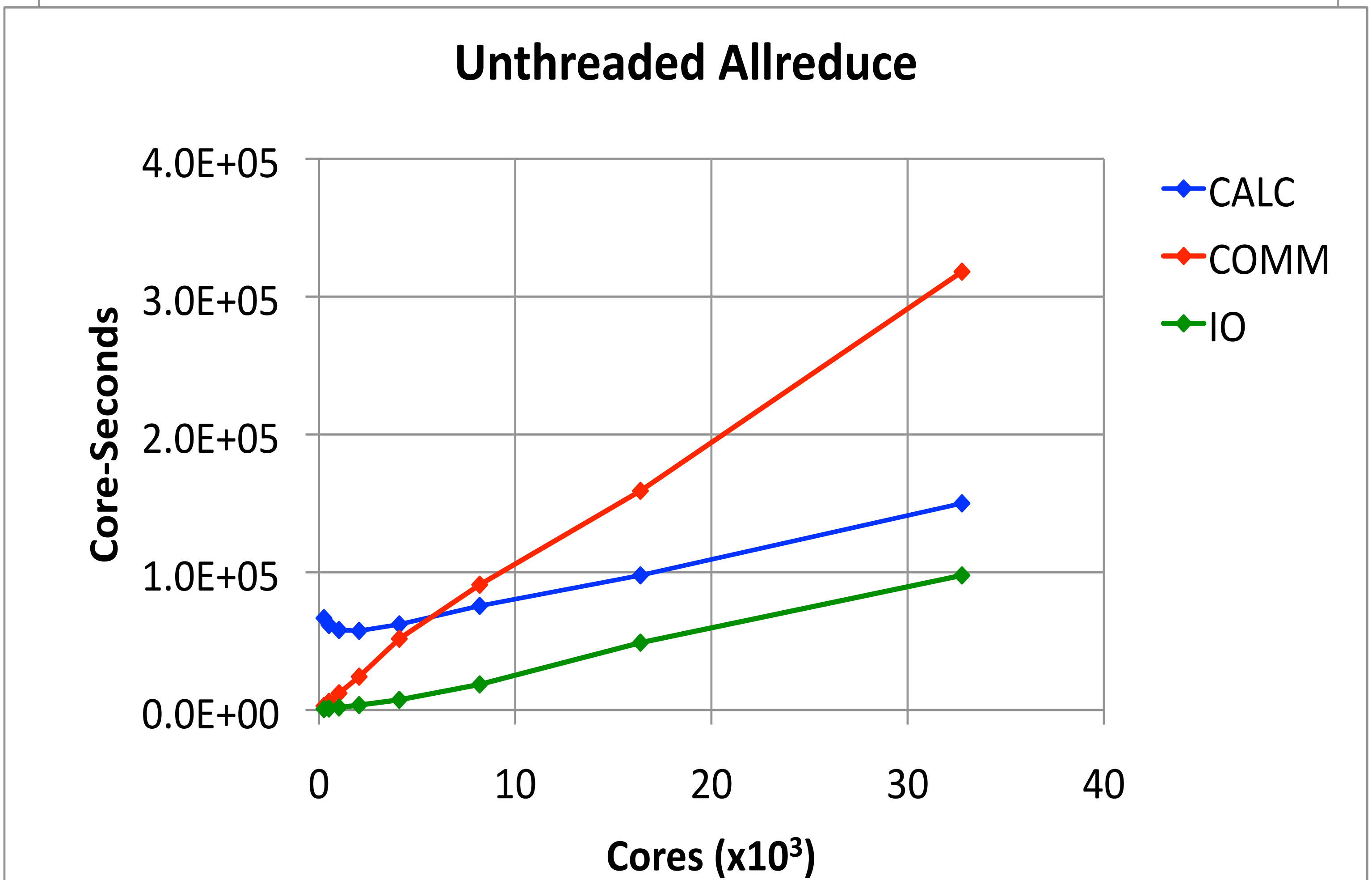
Data volume & computational capability dictate analysis approach.

Date	Data	System	Map	Power Spectrum	
1997 - 2000	B98	Cray T3E x 700	Explicit Maximum Likelihood (Matrix Invert - N_p^3)	Explicit Maximum Likelihood (Matrix Cholesky + Tri-solve - N_p^3)	Algorithms
2000 - 2003	B2K2	IBM SP3 x 3,000	Explicit Maximum Likelihood (Matrix Invert - N_p^3)	Explicit Maximum Likelihood (Matrix Invert + Multiply - N_p^3)	
2003 - 2007	Planck SF	IBM SP3 x 6,000	PCG Maximum Likelihood (band-limited FFT – few N_t)	Monte Carlo (Sim + Map - many N_t)	
2007 - 2010	Planck AF EBEX	Cray XT4 x 40,000	PCG Maximum Likelihood (band-limited FFT – few N_t)	Monte Carlo (SimMap - many N_t)	Implementations
2010 - 2014	Planck MC PolarBear	Cray XE6 x 150,000	PCG Maximum Likelihood (band-limited FFT – few N_t)	Monte Carlo (Hybrid SimMap - many N_t)	

COMPUTATIONAL COSMOLOGY CENTER

BERKELEY LAB

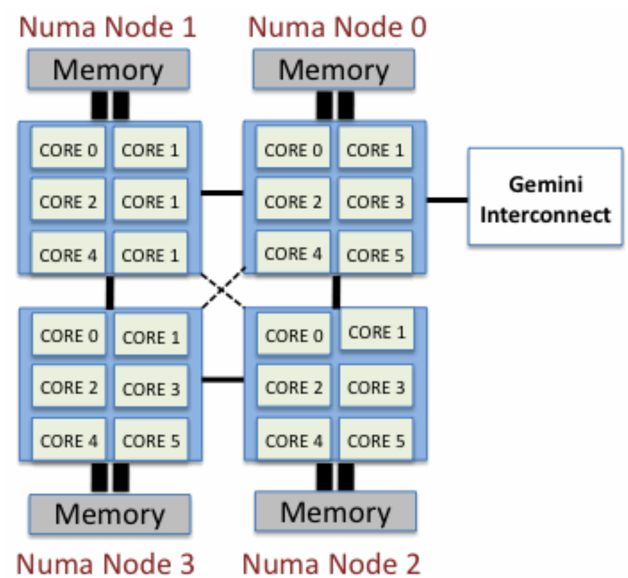
Scaling



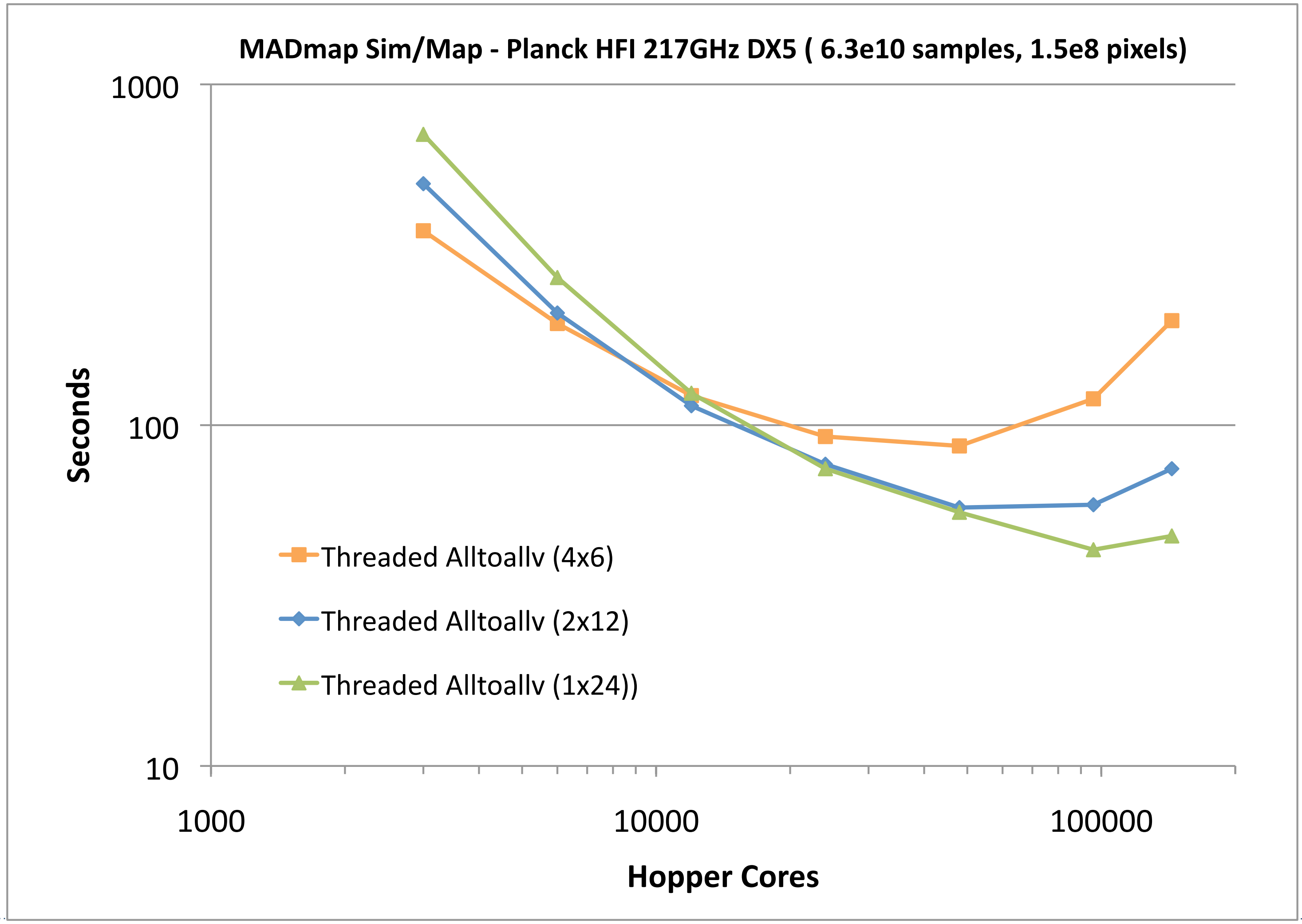
Unthreaded Allreduce
4.0E+05
3.0E+05
2.0E+05
1.0E+05
0.0E+00
Core-Seconds
0
10
20
30
40
Cores (x10³)
CALC
COMM
IO

Heterogeneous HPC Systems

- Clock speed is no longer able to maintain Moore's Law.
  - multi-core CPU, GPGPU, …
- E.g. NERSC's new XE6 system *Hopper*
  - 6384 nodes
  - 2 Magny Cours processors per node
  - 2 NUMA nodes per processor
  - 6 cores per NUMA node
- What is the best way to run hybrid code on such a system?
  - "wisdom" says 4 processes x 6 threads to avoid NUMA effects.



NUMA vs MPI



MADmap Sim/Map - Planck HFI 217GHz DX5 (6.3e10 samples, 1.5e8 pixels)
1000
100
10
Seconds
1000
10000
100000
Hopper Cores
Threaded Alltoallv (4x6)
Threaded Alltoallv (2x12)
Threaded Alltoallv (1x24))

Conclusions

- Not all data are images; not all projects end up in a database!
- Data volumes require algorithms with minimal data re-use
  – no room to hide computational inefficiencies.
- Hierarchy of (time, power) costs drives implementation approach
  – cost ratios get worse with concurrency/generation.
- Heterogeneous/hierarchical architectures add an additional layer (or more) of complexity
  – the responsibility to address this lies with us.
  – (how best) can we influence the degree of the challenge?